Abarna_day86

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int compare(const void *a, const void *b) {
    return (*(int *)a - *(int *)b);
}

void rebuild_array(int n, int *subset_sums, int *result) {
    int size = 1 << n;
    int *count = (int *)calloc(size, sizeof(int));
    qsort(subset_sums, size, sizeof(int), compare);

    int result_index = 0;
    for (int i = 1; result_index < n; i++) {
        if (count[i] > 0) continue;

        int value = subset_sums[i];
        result[result_index++] = value;

        for (int j = 0; j < size; j++) {
            if (count[j] == 0 && subset_sums[j] == value) {
                count[j]++;
                value -= subset_sums[j];
            }
            if (value == 0) break;
        }
    }

    free(count);
}

int main() {
    int T;
    scanf("%d", &T);

    while (T--) {
        int N;
        scanf("%d", &N);

        int size = 1 << N;
        int *subset_sums = (int *)malloc(size * sizeof(int));
        for (int i = 0; i < size; i++) {
            scanf("%d", &subset_sums[i]);
        }
```

```
        int *result = (int *)malloc(N * sizeof(int));
        rebuild_array(N, subset_sums, result);

        for (int i = 0; i < N; i++) {
            printf("%d ", result[i]);
        }
        printf("\n");

        free(subset_sums);
        free(result);
    }

    return 0;
}
```